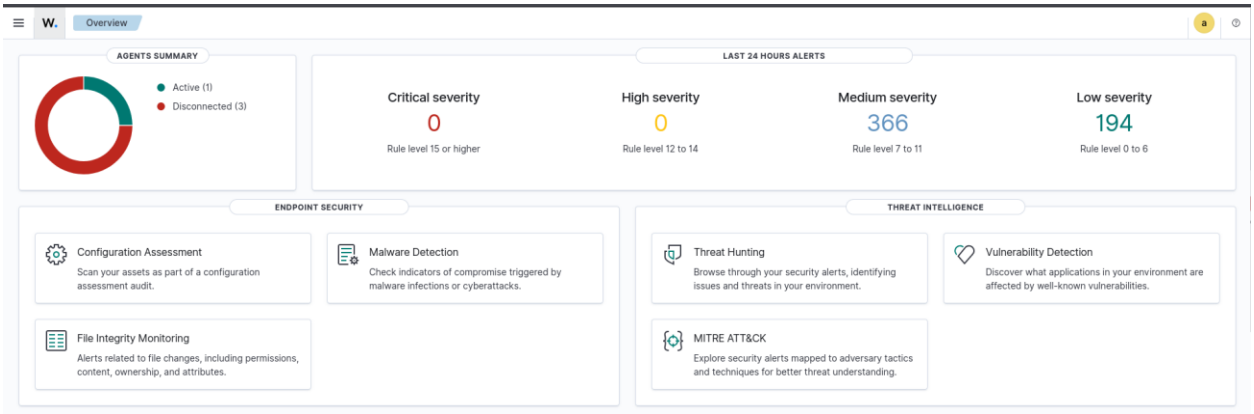
Detecting known bad actors with Wazuh and AbuseIPDB

GitHub: github.com/sameerhassancode/Wazuh-labs

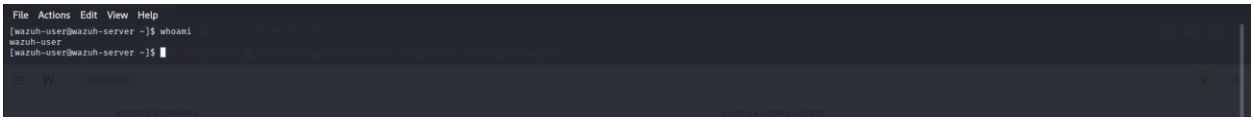
AbuseIPDB is a project that helps systems administrators, webmasters, and security analysts check and report IP addresses involved in various categories of malicious attacks. It provides an API to check and report an IP address for malicious activity.

Wazuh supports integrating with external software using the integrator tool. Integrations are done by connecting the Wazuh manager with APIs of the software products through scripts. We currently support integrations with VirusTotal, Slack, and PagerDuty out of the box, while providing an option for creating custom integrations.

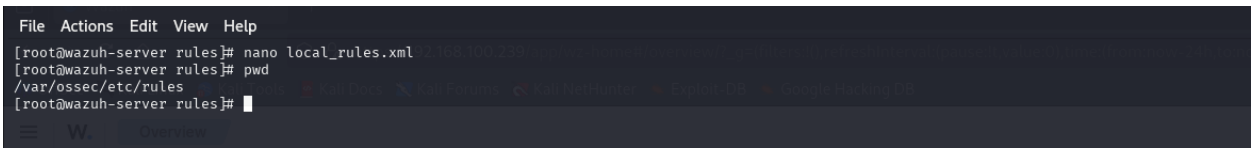
Wazuh-Dashboard:



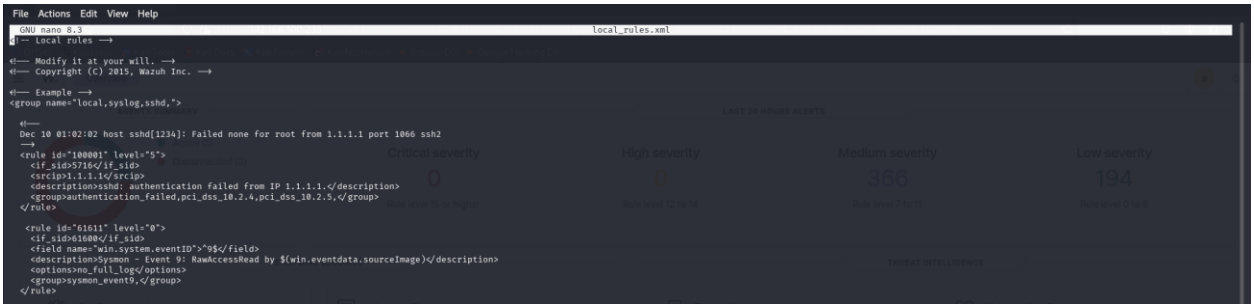
Connect the Wazuh through SSH:



Now Change directory to this path ” `/var/ossec/etc/rules/local_rules.xml`”



Now open the file to add custom rules for AbusePBD.



Scroll down to the end of file. And paste these rules.

[Detecting known bad actors with Wazuh and AbuseIPDB | Wazuh](#) (Copy from here)

```
<group name="local,syslog,sshd,">
  <rule id="100002" level="5">
    <if_sid>5716</if_sid>
    <match type="pcre2">\b(?!(10)|192\.168|172\.(2[0-9]|1[6-9]|3[0-1])|(25[6-9]|2[6-9][0-9]|[3-9][0-9][0-9]|99[1-9]))[0-9]{1,3}\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)</match>
    <description>sshd: Authentication failed from a public IP address $(srcip).</description>

<group>authentication_failed,authentication_success,pci_dss_10.2.4,pci_dss_10.2.5,</group>
  </rule>

  <rule id="100003" level="5">
    <if_sid>5715</if_sid>
    <match type="pcre2">\b(?!(10)|192\.168|172\.(2[0-9]|1[6-9]|3[0-1])|(25[6-9]|2[6-9][0-9]|[3-9][0-9][0-9]|99[1-9]))[0-9]{1,3}\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)</match>
```

<description>sshd: Authentication succeeded from a public IP address $(srcip).</description>

<group>authentication_failed,authentication_success,pci_dss_10.2.4,pci_dss_10.2.5,</group>

</rule>

</group>

Once the rule IDs 100002 or 100003 triggers an alert, the Wazuh integration script makes a request to the AbuseIPDB Check IP API endpoint and returns information about the IP address in the alert. This is subsequently used in a rule created based on the Confidence of Abuse score.

Now Change directory to /var/ossec/etc and open Osse.conf file.

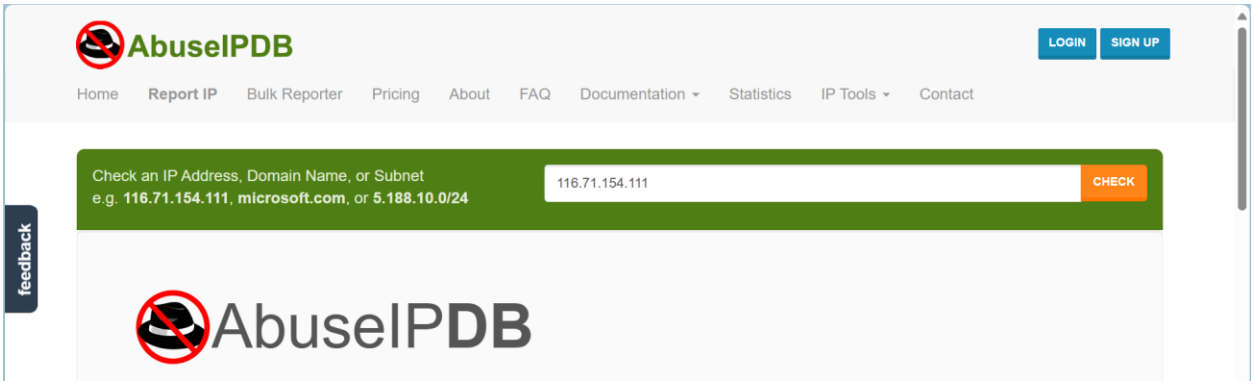
Open ossec.conf file

Now scroll down to Integration Section and paste this code:

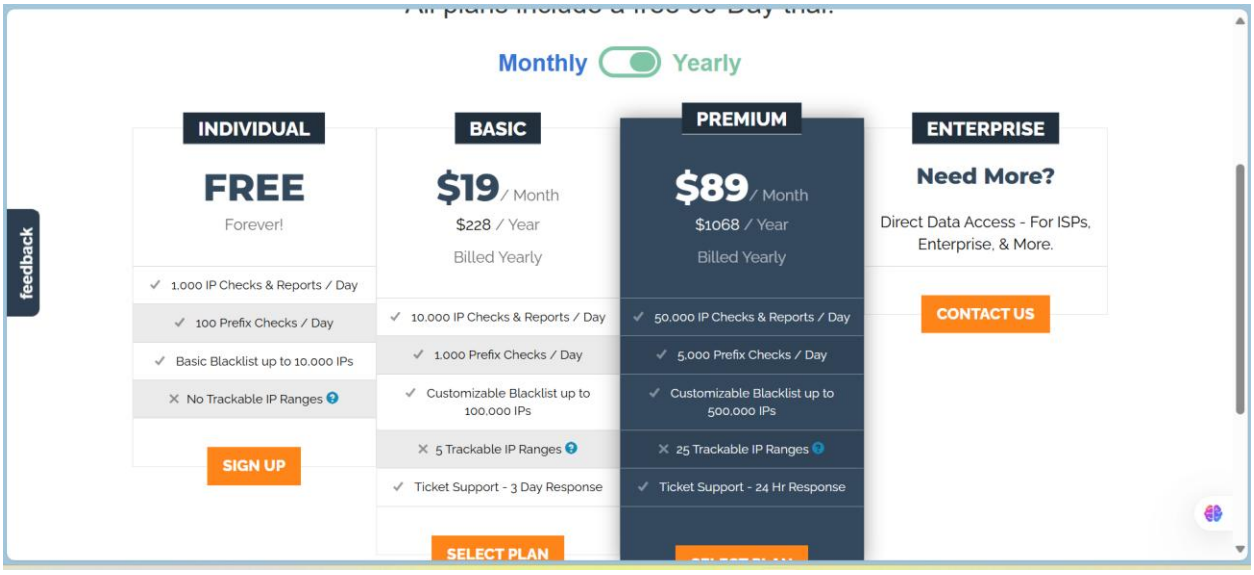
```
<integration>
  <name>custom-abuseipdb.py</name>
  <hook_url>https://api.abuseipdb.com/api/v2/check</hook_url> (add Hook url )
  <api_key><YOUR_ABUSEIPDB_API_KEY></api_key> ( add api key )
```

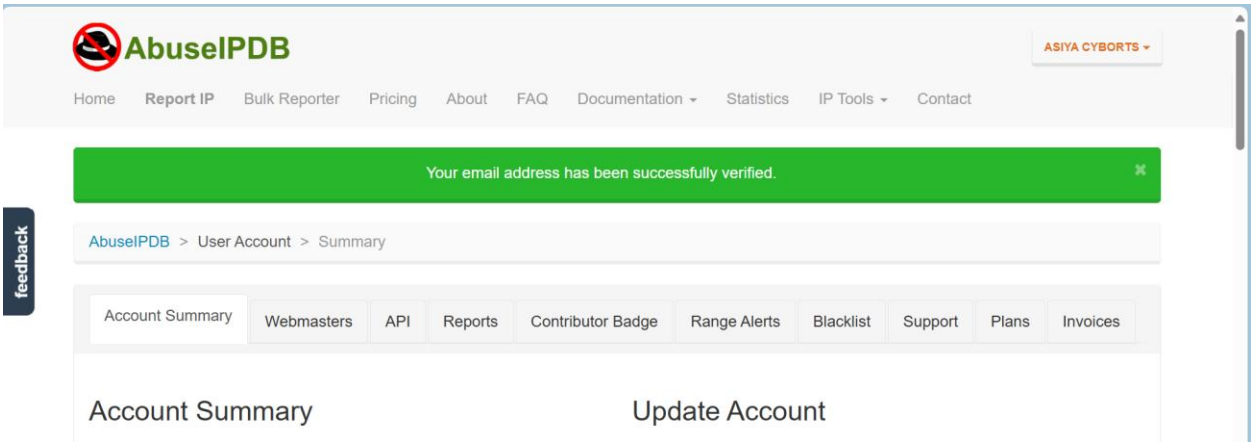
```
  <rule_id>100002,100003</rule_id>
  <alert_format>json</alert_format>
</integration>
```

To get API visit AbusePBD website

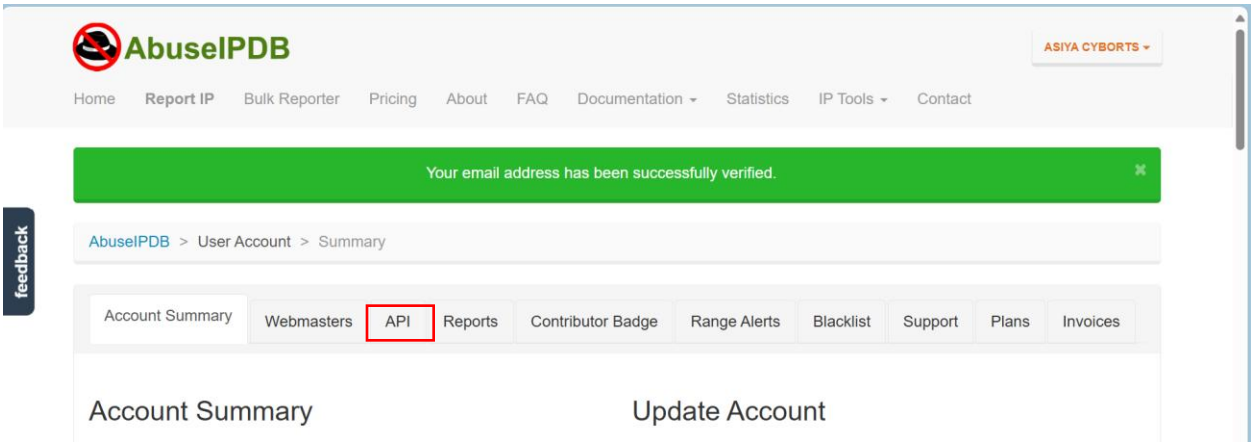


Now click on SIGNUP button and then select your subscription plan and click on SignUp

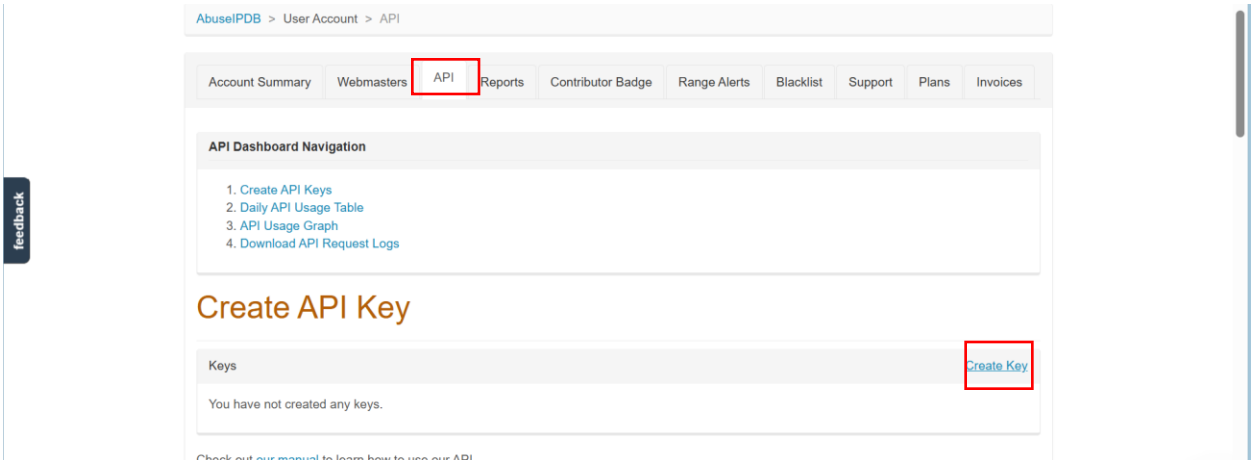




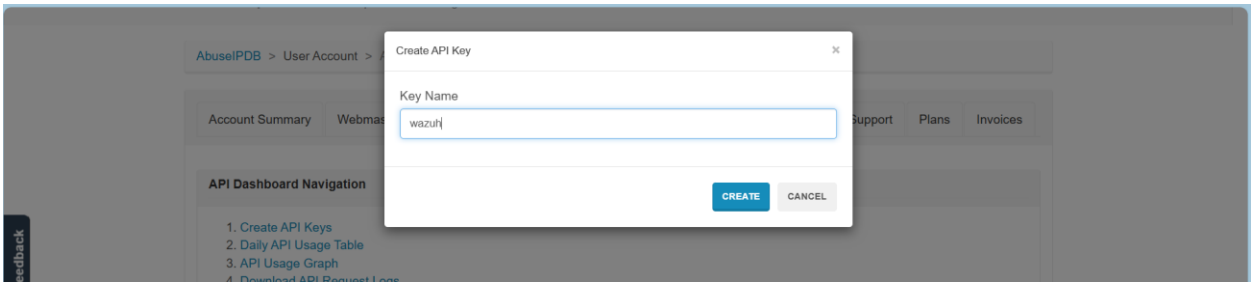
After Login Clcik on API tab



Now Click on Create Key



Type Key name and click create



Now, Key is ready to import copy the key and paste it under api_key tag


```
  <!-- Virustotal Inventory -->
  <integration>
    <name>virustotal</name>
    <api_key>56993d1619558[redacted]</api_key>
    <group>syscheck</group>
    <alert_format>json</alert_format>
  </integration>

<integration>
  <name>custom-abuseipdb.py</name>
  <hook_url>https://api.abuseipdb.com/api/v2/check</hook_url>
  <api_key>c5cd497c941[redacted]x</api_key>
  <rule_id>100002,100003</rule_id>
  <alert_format>json</alert_format>
</integration>
```

Now save the file and change directory to /var/ossec/integration folder.

Create a new file using command touch **filename.**

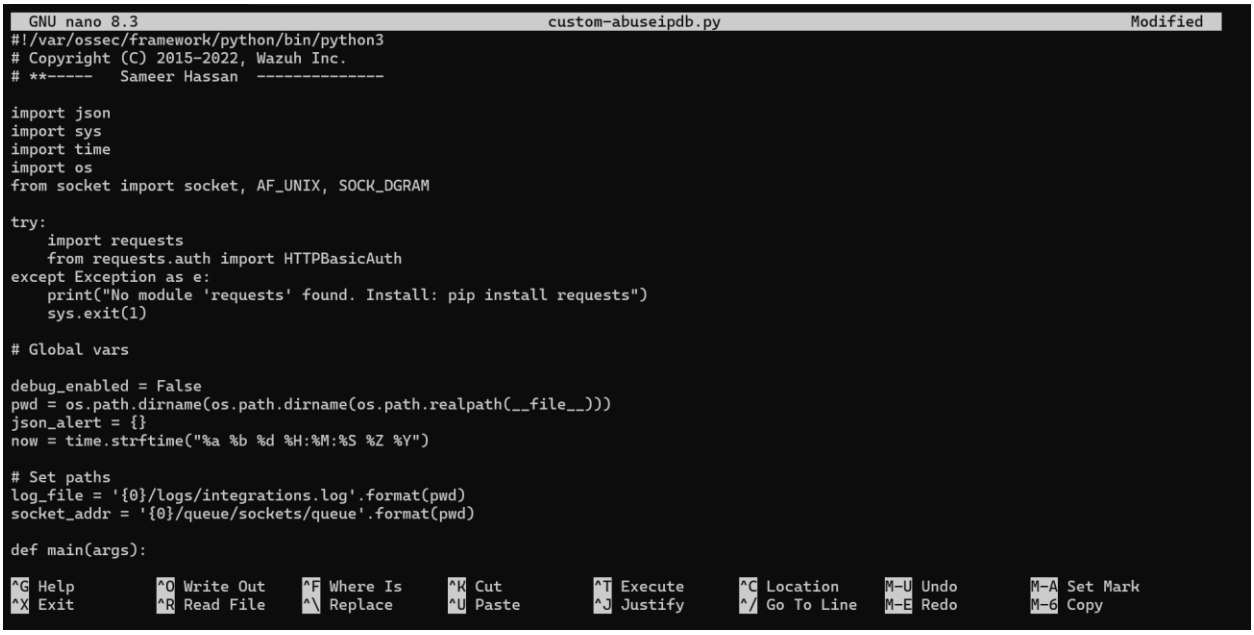
```
[root@wazuh-server integrations]# pwd
/var/ossec/integrations
[root@wazuh-server integrations]# ls
maltiverse  maltiverse.py  pagerduty  pagerduty.py  shuffle  shuffle.py  slack  slack.py  virustotal  virustotal.py
[root@wazuh-server integrations]# touch custom-abuseipdb.py
[root@wazuh-server integrations]# ls
custom-abuseipdb.py  maltiverse.py  pagerduty.py  shuffle.py  slack.py    virustotal.py
maltiverse           pagerduty      shuffle       slack       virustotal
[root@wazuh-server integrations]#
```

Open the file with nano **command.**

Copy the code from here and paste it

Link: [Detecting known bad actors with Wazuh and AbuseIPDB | Wazuh]

Save the file



After saving the file. Change the permission of file with given command

```
chmod 750 /var/ossec/integrations/custom-abuseipdb.py
```

```
[root@wazuh-server integrations]# chmod 750 /var/ossec/integrations/custom-abuseipdb.py
[root@wazuh-server integrations]#
```

```
chown root:wazuh /var/ossec/integrations/custom-abuseipdb.py
```

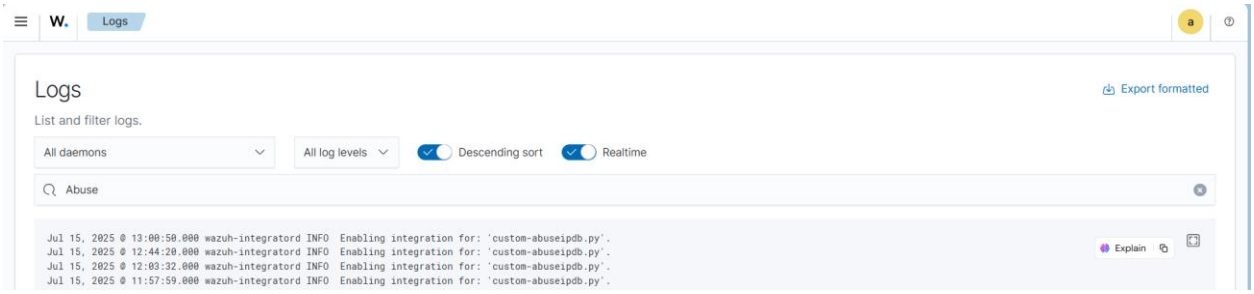
```
[root@wazuh-server integrations]# chown root:wazuh /var/ossec/integrations/custom-abuseipdb.py
```

Now restart the Wazuh-manager.

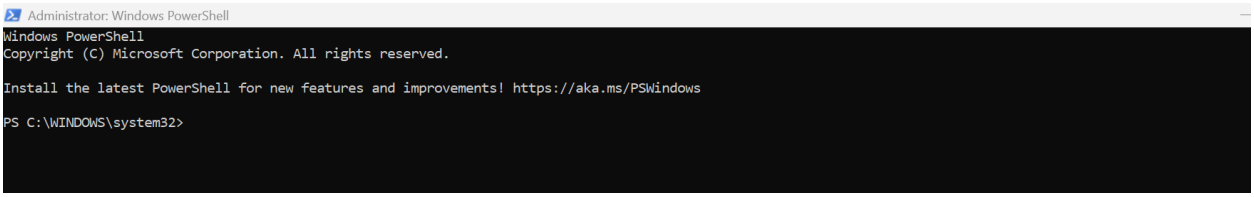
Systemctl restart Wazuh-manager.

```
[root@wazuh-server integrations]# systemctl restart wazuh-manager
```

AbusePBD is working properly now lets check it by generating some alerts.



Now start the PowerShell with Admin Privileges

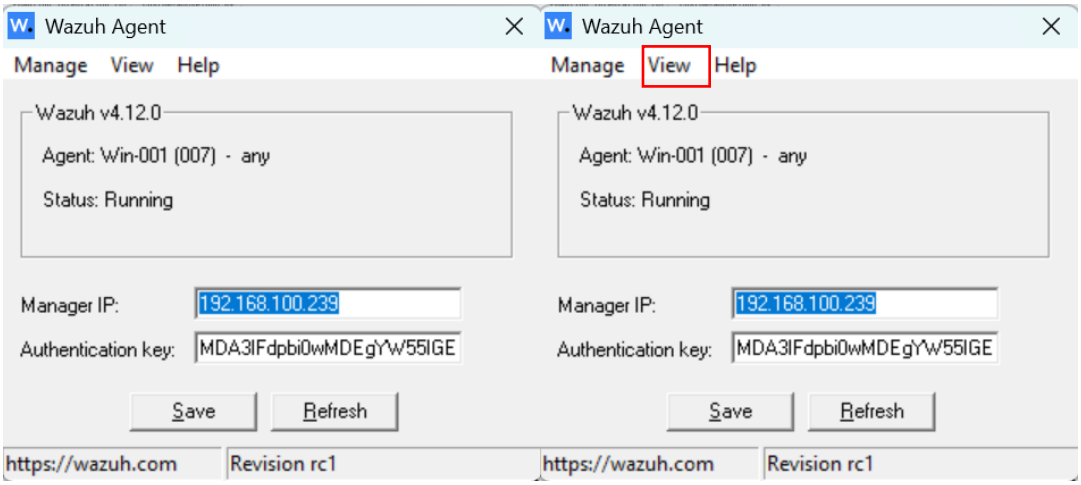


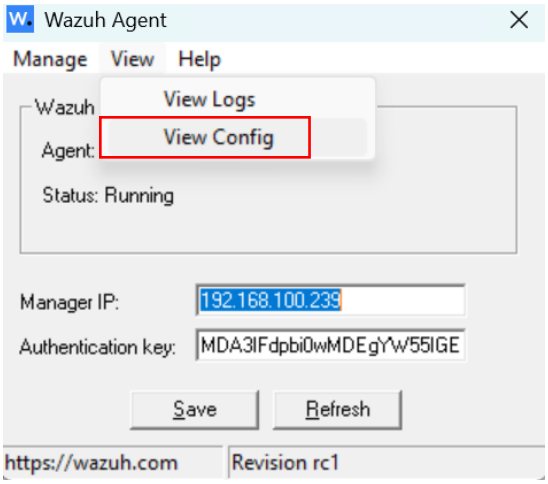
Run this command to add test logs

Command

```
Add-Content -Path "C:\Program Files (x86)\ossec-agent\logs\test_abuse_log.log" `
        -Value "Suspicious connection attempt to 45.147.229.180"
```

Now open the agent configuration file then click on **view** then **view config**





After opening the file goto the local file section and add the path of file.

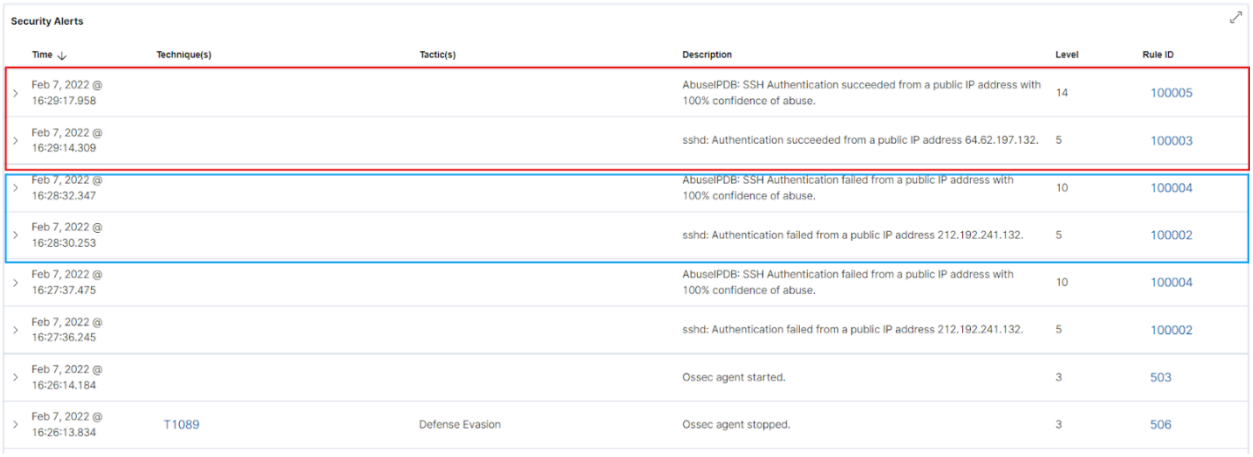
```
  <localfile>
    <location>System</location>
    <log_format>eventchannel</log_format>
  </localfile>

<localfile>
  <location>C:\Temp\wazuh_test.log</location>
  <log_format>syslog</log_format>
</localfile>

  <localfile>
    <location>Microsoft-Windows-Sysmon/Operational</location>
    <log_format>eventchannel</log_format>
  </localfile>

  <localfile>
```

Now restart the Wazuh agent and then go to Dashboard.



Security Alerts

Time	Technique(s)	Tactic(s)	Description	Level	Rule ID
Feb 7, 2022 @ 16:29:17.958			AbuseIPDB: SSH Authentication succeeded from a public IP address with 100% confidence of abuse.	14	100005
Feb 7, 2022 @ 16:29:14.309			sshd: Authentication succeeded from a public IP address 64.62.197.132.	5	100003
Feb 7, 2022 @ 16:28:32.347			AbuseIPDB: SSH Authentication failed from a public IP address with 100% confidence of abuse.	10	100004
Feb 7, 2022 @ 16:28:30.253			sshd: Authentication failed from a public IP address 212.192.241.132.	5	100002
Feb 7, 2022 @ 16:27:37.475			AbuseIPDB: SSH Authentication failed from a public IP address with 100% confidence of abuse.	10	100004
Feb 7, 2022 @ 16:27:36.245			sshd: Authentication failed from a public IP address 212.192.241.132.	5	100002
Feb 7, 2022 @ 16:26:14.184			Ossec agent started.	3	503
Feb 7, 2022 @ 16:26:13.834	T1089	Defense Evasion	Ossec agent stopped.	3	506

Conclusion:

I integrated AbuseIPDB API with Wazuh to check IP addresses associated with malicious activity. This integration allowed us to retrieve information from AbuseIPDB about public IP addresses that attempted SSH authentication. The information retrieved was subsequently used with rules to improve the detection of known bad actors.